

CLAIM AMENDMENT

Please amend the claims as follows:

1. (Canceled)
2. (Canceled)
3. (Canceled)
4. (Canceled)
5. (Canceled)
6. (Canceled)
7. (Canceled)
8. (currently amended) A method for using a first computer system to remotely monitor and interact with the operation of a second computer system through a graphical user interface of said second computer system, comprising the steps of:

receiving a pixel image of a first graphical element of said second computer system graphical user interface at said first computer system;

generating a user peripheral input device input action within said second computer system graphical user interface as interpreted by a second computer peripheral input device controller channel by passing a signal through an i/o communications channel from said first computer system to said second computer system graphical user interface responsive to said receiving step;

monitoring a pixel image of said second computer system graphical user interface from said first computer system for an expected second graphical element within a predetermined time interval; and

signaling a failure at said first computer system if said predetermined time interval elapses without detecting said expected second graphical element.

9. (Previously presented) The method of claim 8 further comprising the steps of:
 - transferring said user input action to a script stored on said first computer system;
 - re-executing said steps of receiving, generating, monitoring and signaling subsequent to said storing step under control of said stored script.
10. (Previously presented) The method of claim 8 further comprising the steps of:
 - providing graphical user interface language extensions commands to a scripting

language; and

passing said generated user input action through said graphical user interface language extensions from a scripting language processor to a language extensions processor.

11. (Previously presented) The method of claim 8 further comprising the steps of:

generating a user input action within said second computer system responsive to said second graphical element;
monitoring said second computer system graphical user interface for an expected third graphical element within a predetermined time interval; and
signaling a failure at said first computer system if said predetermined time interval elapses without detecting said expected third graphical element.

12. (Previously presented) The method of claim 8 further comprising the steps of:

depicting said second computer system graphical user interface upon a local display of said first computer system including said first graphical element; and
receiving a local user input action at said first computer system within said local display;
wherein said generated user input action emulates said local user input action.

13. (Previously presented) The method of claim 8 further comprising the steps of:

providing graphical user interface language extensions commands to a scripting language; and

depicting said computer system graphical user interface upon a local display of said first computer system including said first graphical element;

receiving a local user input action within said local display;

transferring said user input action to a script stored on said first computer system;

passing said generated user input action through said graphical user interface language extensions from said a scripting language processor to a language extensions processor for reproduction at said second computer system graphical user interface, wherein said generated user input action emulates said local user input action; and

re-executing said steps of receiving, generating, monitoring and signaling subsequent to said storing step under control of said stored script.

14. (currently amended) A method for enabling a local system to remotely operate a remote computer system through a graphical user interface on said remote computer system by using local scripts that selectively respond to changes in graphical displays upon said graphical user interface of said remote computer system, comprising the steps of:

displaying a depiction of said remote system graphical user interface display on said

local system;
capturing user input effected in said depiction of said remote system graphical user interface display;
implementing through a local system command language set user input emulations representative of said captured user input reproduced at said remote computer system graphical user interface through a peripheral input device i/o channel;
image processing said remote computer system graphical displays to detect changes in said graphical display upon said graphical user interface of said remote computer system;
controlling a flow of execution of said local system through a scripting language having scripting commands in combination with said command language set responsive to a detection of changes during said image processing step; and
communicating between said local system and said remote computer system graphical user interface through a communication interface responsive to said flow controlling step.

15. (currently amended) The method for enabling a local system to remotely operate a remote computer system through a graphical user interface on said remote computer system of claim 14 further comprising the steps of:

storing said scripting commands into a storing means;

inserting a command from said command language set into said storing means; and
executing said inserted stored command.

16. (New) A method for using a first computer system to remotely monitor and interact with the operation of a second computer system through a graphical user interface of said second computer system, comprising the steps of:

receiving a first graphical element of said second computer system graphical user interface at said first computer system;
generating a user peripheral input device input action within said second computer system graphical user interface as interpreted by a second computer peripheral input device controller channel by passing a signal through an i/o communications channel from said first computer system to said second computer system graphical user interface responsive to said receiving step;
monitoring said second computer system graphical user interface from said first computer system for an expected second graphical element within a predetermined time interval; and
signaling a failure at said first computer system if said predetermined time interval elapses without detecting said expected second graphical element.

17. (New) The method of claim 16 further comprising the steps of:

transferring said user input action to a script stored on said first computer system;
re-executing said steps of receiving, generating, monitoring and signaling subsequent
to said storing step under control of said stored script.

18. (New) The method of claim 16 further comprising the steps of:

providing graphical user interface language extensions commands to a scripting
language; and
passing said generated user input action through said graphical user interface
language extensions from a scripting language processor to a language
extensions processor.

19. (New) The method of claim 16 further comprising the steps of:

generating a user input action within said second computer system responsive to said
second graphical element;
monitoring said second computer system graphical user interface for an expected
third graphical element within a predetermined time interval; and
signaling a failure at said first computer system if said predetermined time interval
elapses without detecting said expected third graphical element.

20. (New) The method of claim 16 further comprising the steps of:

depicting said second computer system graphical user interface upon a local display
of said first computer system including said first graphical element; and
receiving a local user input action at said first computer system within said local
display;
wherein said generated user input action emulates said local user input action.

21. (New) The method of claim 16 further comprising the steps of:

providing graphical user interface language extensions commands to a scripting
language; and
depicting said computer system graphical user interface upon a local display of said
first computer system including said first graphical element;
receiving a local user input action within said local display;
transferring said user input action to a script stored on said first computer system;
passing said generated user input action through said graphical user interface
language extensions from said a scripting language processor to a language
extensions processor for reproduction at said second computer system
graphical user interface, wherein said generated user input action emulates
said local user input action; and
re-executing said steps of receiving, generating, monitoring and signaling subsequent
to said storing step under control of said stored script.